Synopsis

SYNOPSIS

Laptops have become a valuable part of the computing arsenal. They allow users powerful mobile computers with the same capacity and software of many desktops.

Asset Tracking System uses the web application to help us track each computer, where it is located, what is status of laptop, who is repairing in and, should one go missing, where it has been taken. We just report the missing computer to absolute and they work with local law enforcement to find it and recover it for us. We can also inventory our computers at any time.

Asset Tracking System is extremely valuable for laptop repairing company who repairing the laptop and what is the status of the laptop. Unfortunately, the mobility, technology and information that make laptops so useful to employees and organizations also makes them valuable prizes for thieves. This article, the first in a two-part series devoted to laptop security, will give a brief overview of how users can prevent laptop theft.

Introduction

INTRODUCTION

Asset Tracking System is used to locating or tracking static or in-motion assets, such as laptops, healthcare facility, servers in a data center or laptops in a corporation, was no easy task.

Active RFID allows users to instantly determine the general location of tagged assets anywhere within the facility. Installation of "control point" Detection zones at strategic locations throughout the facility allows the user to define logical zones and monitor high traffic areas. Tagged assets moving through these control points provide instant location data.

RFID Asset Tags (Applied to the items)

An Asset Tag is a small transponder that remains in a sleep state until activated. When the Tag receives a special Very Low Frequency (VLF) signal from the Activator, the Tag wakes up and emits an Ultra High Frequency (UHF) radio signal. The Asset Tag can be mounted to an asset for effective protection and monitoring.

RFID Activator (wakes up the Tag)

The Activator broadcasts an activation signal via the transmit antenna to wake up the tag. The transmitting antenna attaches to the BNC connector on top of the Activator. The wakeup signal writes the Activators ID Number to the tag.

Asset Tag Receiver (stand alone - connects to web)

The Receiver decodes the signal from the Tags and converts the data into SQL Server. Some receivers have output control signals for activation TTL outputs or alarm points.The Receiver can be configured to connect to an IP network using an

integral Network Interface Unit (NIU) to convert data from serial to Ethernet TCP/IP format.

Asset Tracking System with solid Database Design, Easy-to-Use, Set of Workflow status for laptop, Supports Passive and Active Radio Frequency Identification (RFID), Active Directory.

The main requirement of the client is to maintain all the tag activities and workflow status of the laptop and alert notification to fetch the required information from the system. The Asset Tracking System includes the entire module such as

- Customer Module
    - Customer Entry
    - Edit Customer
    - Delete Customer
    - Customer List
- Additional Work
- Work Order
- Search
    - Customer Wise
    - Tag Wise
- Notification Log
- Working Tag
    - Tag Entry
    - Active Tag List
- Technician Info
    - Customer Entry

Software Project Plan

SOFTWARE PROJECT PLAN

This chapter discusses about the time schedule for the project, and it contains the various phases of the project.

Time Schedule for Various Phases

S.No	Task	Duration
1.	Requirement Specification	5 days
2.	Requirement Document Specification	5 days
3.	Design Analysis	10 days
4.	Design Documentation	10 days
5.	Design Review	20 days
6.	Coding	20 days
	Total Days :	**70 days**

Customer Requirements Determination

CUSTOMER REQUIREMENTS DETERMINATION

Existing System

Next level of chatting is User-friendly chat. In addition to previous functionality, this provides active users list. Quantity of users on start page is also required. It has logoff possibility and expiration of inactive users after specified timeout.

The main principles are similar, with storing messages from previous example. To generalize tasks, active user term is replaced with user session. User session, for this type of process, should basically contain a unique session identifier and the user name.

The major method underwent in the system for specifying problems were:

- Information Gathering
- Discussion Information's were collected from the user regarding the flow of the system, the design of the system, validation checking and features, which are to be incorporated in system.
- Discussions were carried out with the management to choose the best package for developing the system.

Drawbacks

- Laptop theft
- Knowledge of Current Workflow status
- Accuracy cannot be achieved.
- The manual system is too slow.
- The data stored in the files may not be up-to-date.
- Security facilities are not available.
- It is tedious to update and correct the information in files.
- Preparation of reports is a time consuming task
- Possibility for any unauthorized person entering wrong Details.
- Information is not stored for future reference.
- Systems need a lot of manual calculations.
- Changes of damages of papers containing the Information.
- Difficulty in preparing reports in the present system.
- Less accuracy.
- Chances of redundancy

PROPOSED SYSTEM

The proposed system will maintain all the automatically recording everything from RF Tag including the laptop location, current workflow status and now of laptop is repairing and working in progress.

In addition to that, the above said difficulties are overcome in the proposed system.

Every transaction was saved as the database and retrieve the various report from the database. The Tag Moving log is classified by date. With our strong search engine, you can easily and accurately find the logs you want.

The proposed system will keep track of all the user information that attempts to use the system and to change any information. At the same time some of the important fields and the values are freezed, show that is secured.

Features

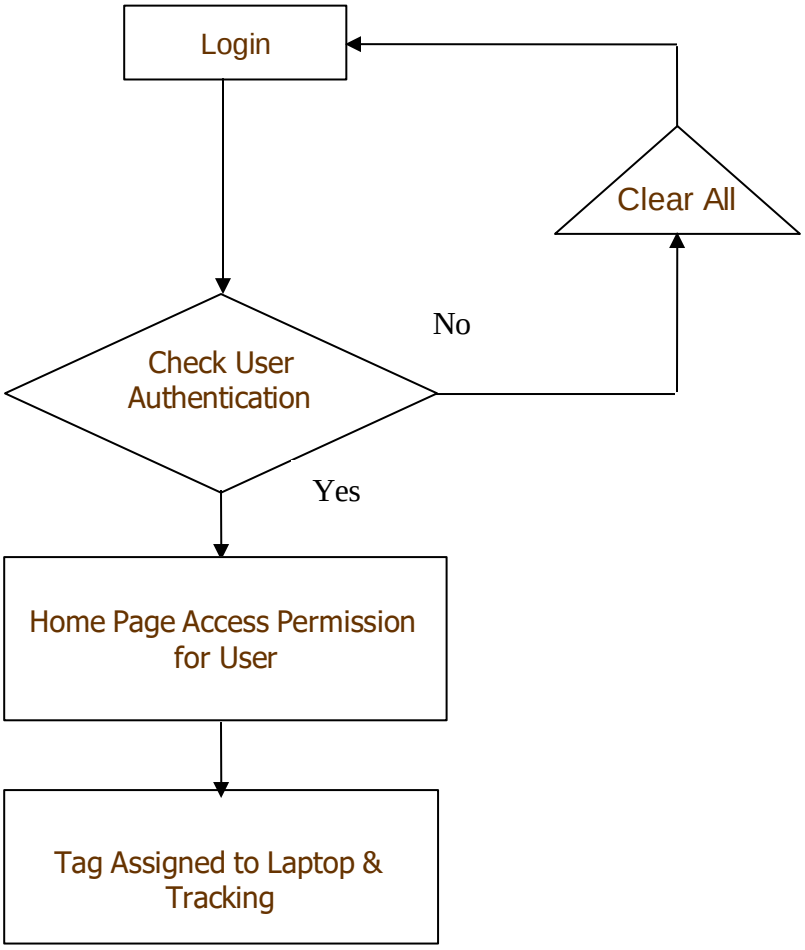
- Maintain the customer information
- Audit the current workflow status
- Automatic online update
- Graphical representation for work flow status
- Automatically sent the SMS and Email to the Customer.
- High reliability, scalability and high load support.

Software Requirements Specification

FUNCTIONAL REQUIREMENTS

A functional requirement defines a function of a software-system or its component. A function is described as a set of inputs, the behavior, and outputs.

Flow Chart



PERFORMANCE REQUIREMENT

The project must satisfy the End user requirements. Accuracy and fast must be imposed in this project.

The project is developed as easy as possible for the sake of end user. The project has to be developed with a view of satisfying the future requirements and future enhancement.

The Asset Tracking System has been finally implemented satisfying the needs specified by the company. As per as the performance is concerned this system said is performing well. The processing time as well as time taken to generate reports where also very less even when large amount of data was used. The system is designed in such a way that even when large amount of data are used for processing there would less performance degradation.

INTERFACE REQUIREMENTS

During a requirements-gathering session for major new information system, key users asked the attributes of the web-oriented graphical interface.

Most interface constraints and restrictions that are imposed by a designer are intended to simplify the mode of interaction.

The user interface should move into the virtual world of the application. The user should not be aware of the operating system, file management functions, or other arcane computing technology .A user should never be required to type operating system commands from within Web application software.

The user feels a sense of control when able to manipulate the objects that are necessary to perform a task in a manner similar to what would occur if the object were physical thing.

OPERATIONAL REQUIREMENTS

The database or databases that are being failed over to the standby server cannot be used for anything else. But databases on the standby server not being used for failover can still be used normally.

When it comes time for the actual failover, you must do one of two things to make your applications work: either rename the standby server the same name as the failed production server (and the IP address), or re-point your user's applications to the new standby server. In some cases, neither of these options is practical.

RESOURCE REQUIREMENTS

Software Requirements:

Server Side Programming	:	ASP.NET.
Middleware Programming	:	Visual Basic
Operating System	:	Windows XP Professional (Service Pack 2)
Web Server	:	Internet Information Scripting Languages
Client Script	:	VB Script, Java Script
Database	:	SQL-Server 2005

Hardware Requirements:

Processor	:	Intel Pentium IV Dual Core 2.8 GHz
Hard Disk	:	80 GB
Monitor	:	LG 17” Color Monitor
RAM	:	1024 MB DDR RAM
Keyboard	:	104 Keys Multimedia Keyboard
Mouse	:	Logitech Optical Mouse
CD – ROM	:	52X CD-ROM.
Floppy Drive	:	1.44 MB

SECURITY REQUIREMENTS

Web Applications are available via network access, it is difficult, if not possible, to limit the population of End-users who may access the application, In order to protect sensitive content and provide secure modes of data transmission, strong security measures must be implemented throughout the infrastructure that supports a Web Application and within the application itself.

Web Applications have become heavily integrated with critical corporate and government databases. E-commerce applications extract and then store sensitive customer information.

For these and many other reasons, Web Application security is paramount in many situations. The key measure of security is the ability of the Web Application and its server environment to rebuff unauthorized access and/or thwart an outright malevolent attack.

DESIGN REQUIREMENTS

Microsoft Office Visio 2003 SDK, providing a work environment that integrates with Visio. This article provides a step-by-step description of how to create shapes in Shape Studio. It shows how to create a project, add base masters and masters to the project, assign behaviors to the masters, create and assign behavior sets, and then apply, test, and validate those behaviors. It also shows how to create and build a stencil to hold the shapes.

Major Steps in Shape Creation

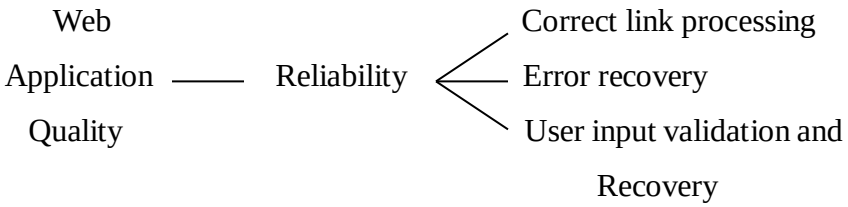
The following list shows the major steps to develop shapes in Shape Studio:

- Create a project
- Create a base master specification
- Add shape geometry to the base master
- Assign behaviors to the base master specification
- Apply base master behavior and validate the base master specification
- Create master specifications based on the base master specification
- Define styles to apply to masters
- Add shape geometry to masters
- Assign behaviors to masters
- Apply master behaviors and validate master specifications
- Create and apply sets of behaviors to masters
- Run developer and test tools on masters
- Create and validate stencil specifications and build stencil files
- Create reports

QUALITY AND RELIABILITY REQUIREMENTS

A software component that is developed for reuse would be verified to be correct and would contain no defects. In reality, formal verification is not carried out routinely, and defects can and do occur. However, with each reuse, defects are found eliminated, and a component's quality improves as a result. Over time, the component becomes virtually defect free.

Software reliability is defined in statistical terms as "the probability of failure – free operation of a computer program in a specified environment for a specified time". The software quality and reliability, failure is nonconformance to software requirements. Failures can be only annoying or catastrophic. One failure can be corrected within seconds while another requires weeks or even months to correct. Complicating the issue even further, the correction of one failure may in fact result in the introduction of the errors that ultimately result in other failures.

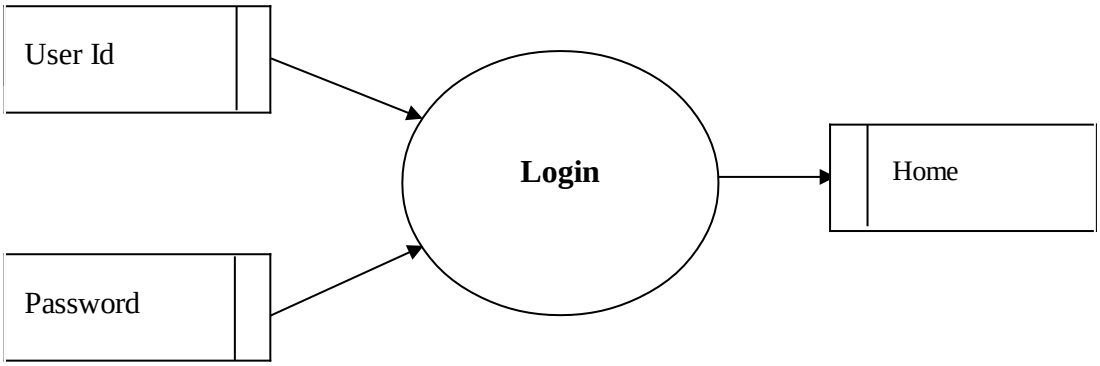


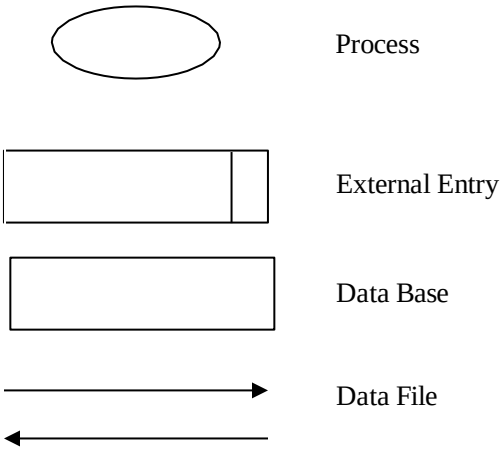
System Analysis

DATA FLOW DIAGRAM

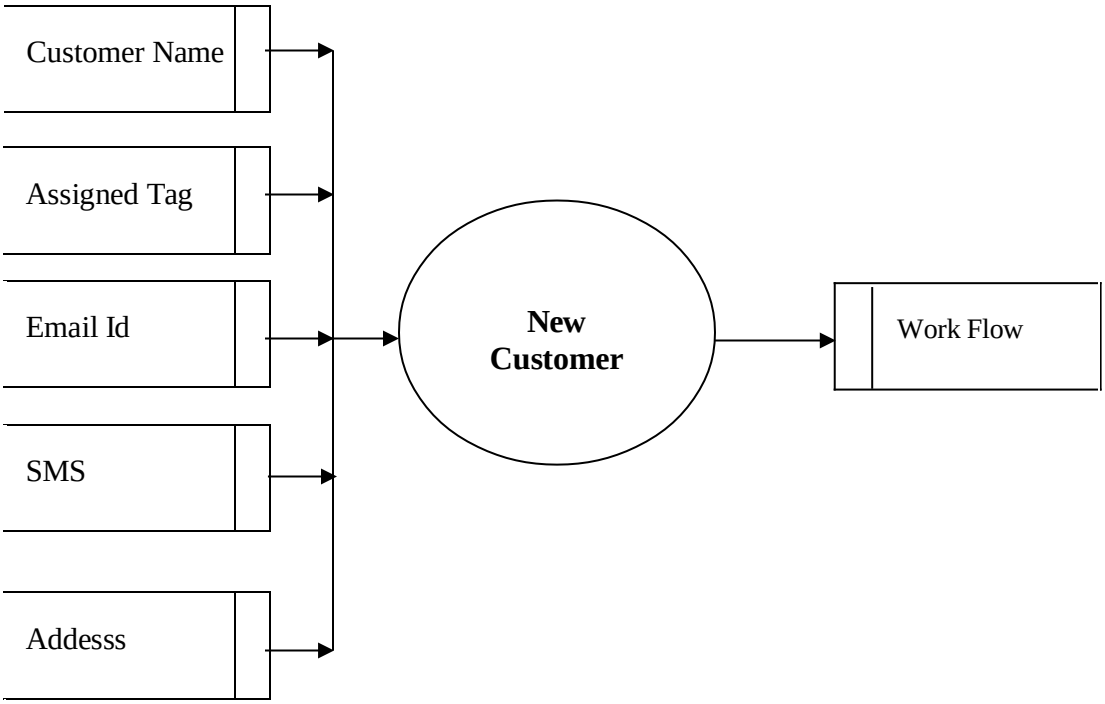
The data flow diagram is one of the most important tools used by system analysis. Data flow diagrams are made up of a number symbols, which represent system components.

Login:

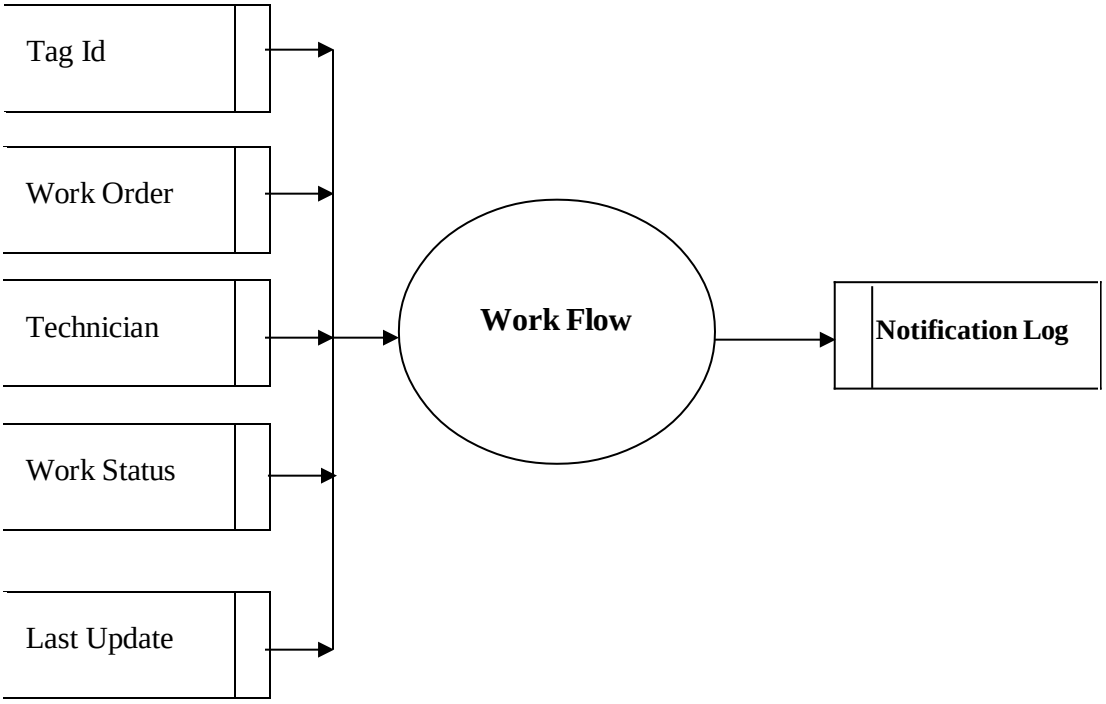




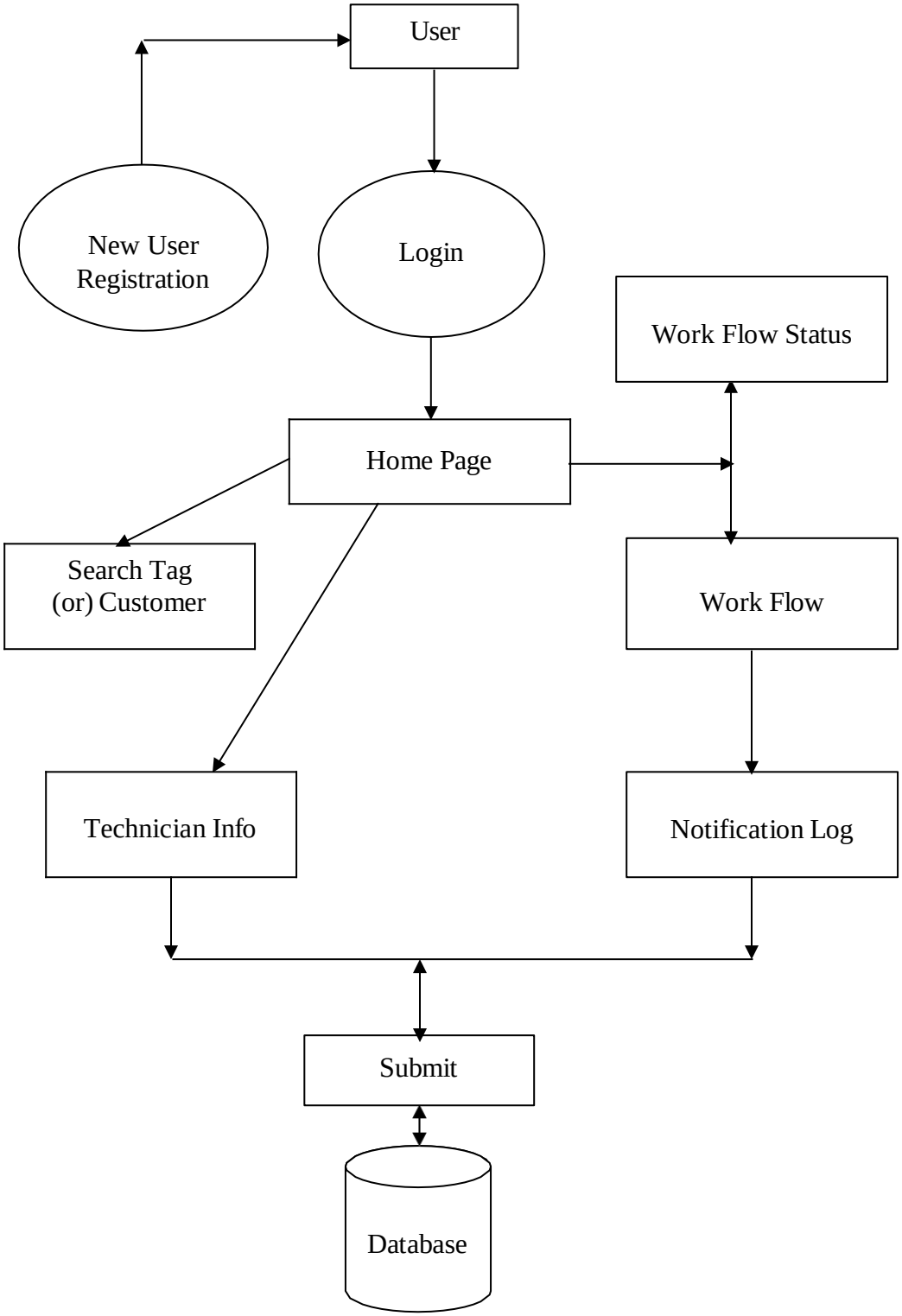
New Customer:



Work Flow



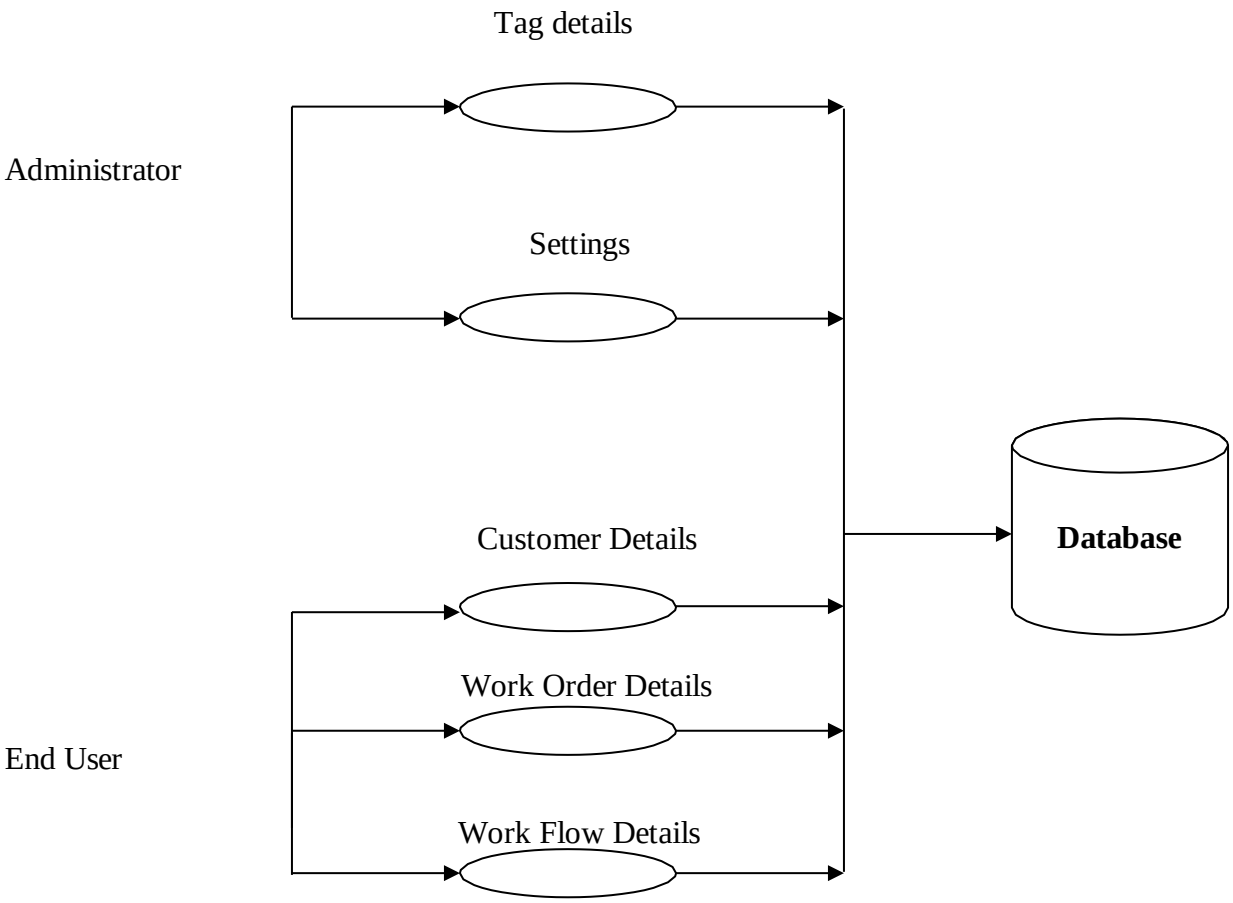
ER DIAGRAM



USE CASE DIAGRAM

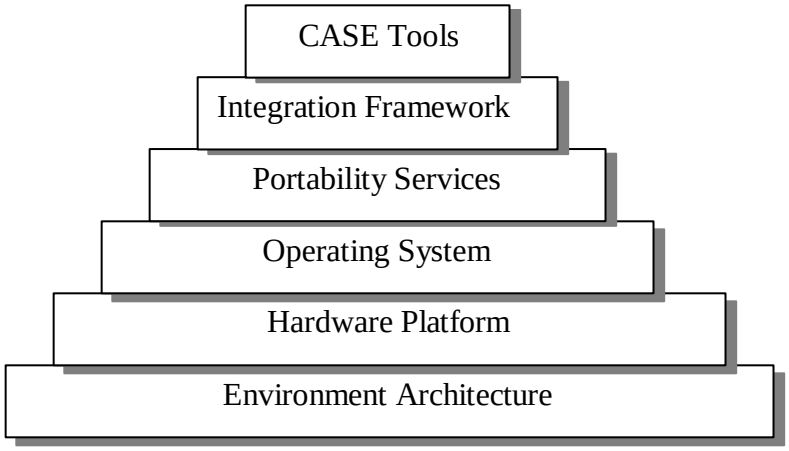
Use Case Diagram

A use case diagram is a type of behavioral diagram defined by the Unified Modeling Language (UML). Its purpose is to present a graphical overview of the functionality provided by a system.



CARE TOOL FOR ANALYSIS

CASE Building Blocks:

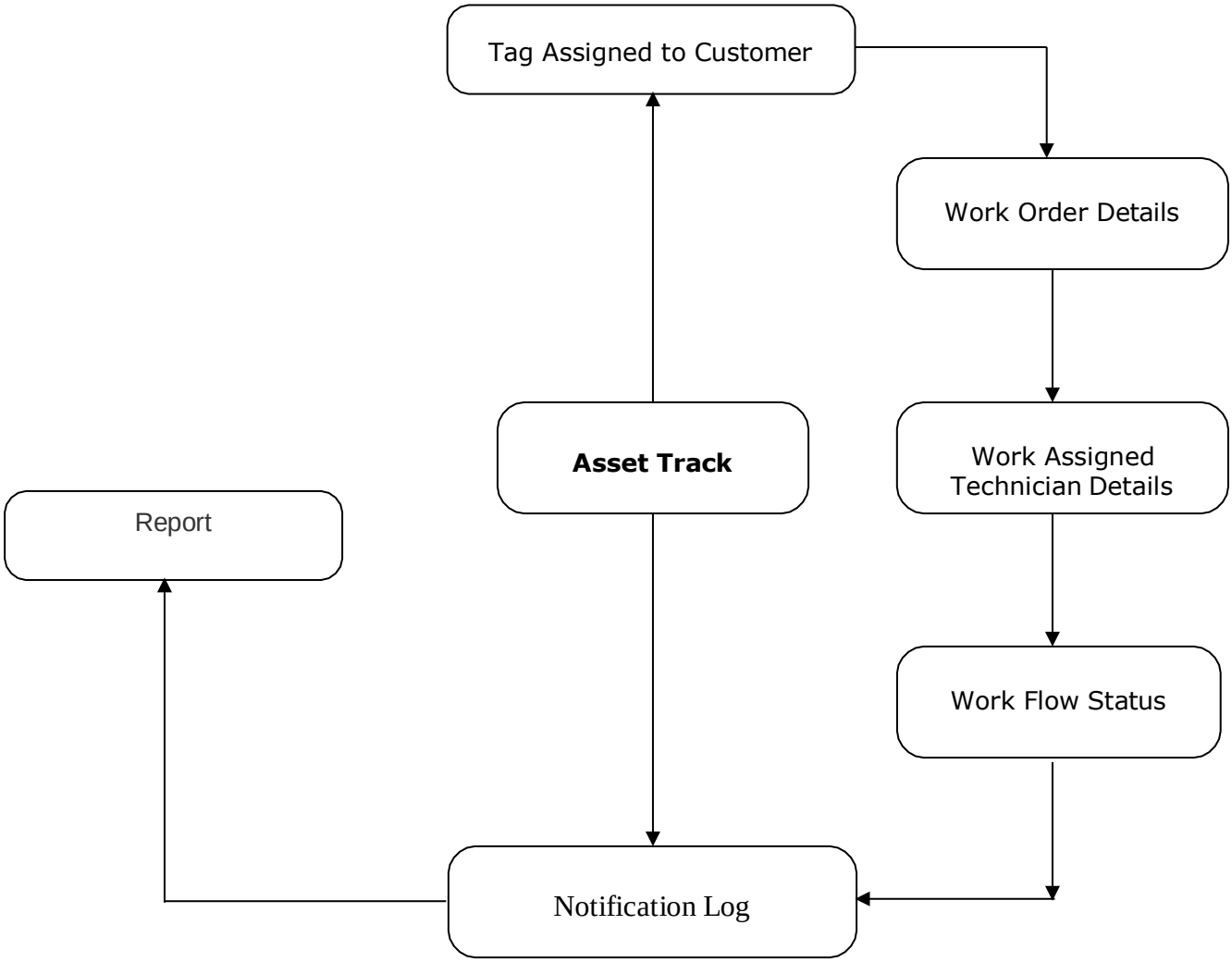


- To test the developed software
- To maintain the implemented software
- To train the new people in software development
- To get clear idea about software engineering processes

The compilers, editors and debuggers those are available to support most conventional programming languages. Web development tools include to the generation of text, graphics, forms, scripts and other elements of a web page.

UML

Unified Modeling Language (**UML**) is a standardized visual specification language for object modeling. UML is a general-purpose modeling language that includes a graphical notation used to create an abstract model of a system, referred to as a UML model.



Design

INPUT DESIGN

In this project, the end user gives the input through the predefined options. For administrator the input is given through the ASP.Net to the database.

In accurate input data is the most common cause of errors in data processing .Input design is the process of converting user oriented inputs into a computer based format. the goal of designing is to make data entry as easy, logical and from errors as possible.

The input forms are designed to be user friendly. Meaningful labels are given to input fields. Immediate validations are done for each input from the user so that the individual inputs are not sending to the gateway.

The input forms are designed so that they provide proper links to the other forms. Each screen has the provision to view the home previous and next page.

The Administrator can create a new user, edit the existing user, delete the existing user and view all user information. But the normal user can only view the user information and edit his / her profile such as First name, last name, and password and so on.

TABLE AND DATABASE DESIGN

A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective of database design is to make the data access easy, inexpensive and flexible to the user.

- In this project, database design is very important.
- The primary key is used in this project.
- It satisfies the Normalization concept.

6.2.1 Normalization:

Thin project consist of second normal form.

Second Normal Form:

A relation is said to be in 2NF if and only if it is in 1NF and every non key attribute is fully functional dependent on the primary key.

The relation users are in 2NF, because every non-key attribute such as Name, Password are fully functional dependent on the primary key User Id.

Table Name : **AlertHistory**

HistoryId	int	
TagId	int	
AlertId	int	
LogTime	datetime	
Synopsis	varchar	250
Name	varchar	100
WorkOrderno	int	

Table Name : **Customers**

CustId	int	
LastName	varchar	50
SMS	varchar	50
EMail	varchar	50
Phone	varchar	50
OtherEmail	varchar	50
Createdon	datetime	
Updatedon	datetime	
FirstName	varchar	50

Table Name : **EventLog**

EventLogId	int
WorkOrderNo	int
StatusId	int
LogTime	datetime
Status	bit

IsSeen	bit	
UserId	int	

Table Name : **NotificationLog**

LogId	int	
EventLogId	int	
ActionId	int	
Remarks	varchar	100
Status	bit	
LogTime	datetime	

Table Name : **NotifyActions**

Action	varchar	100
ActionId	int	

Table Name : **Settings**

AlertTime	int	
EmailServer	varchar	100
SMSServer	varchar	100
AlarmTechCounter	int	
AlarmBackoffice	int	
ChkTagIdFromTagData	bit	

Table Name : **TagData**

DataId	bigint	
TagId	int	
StarId	tinyint	
RoomId	smallint	
Keys	tinyint	
DataIdx	tinyint	
Updatedon	datetime	

Table Name : **TechnicianInfo**

TechId	bigint	
TechName	varchar	100

Table Name : **UnitStatus**

Notify	bit	
StatusId	int	
Description	varchar	100

Table Name : **UserMaster**

UserId	int	
LoginName	varchar	50
Password	varchar	50
Lastname	varchar	50
LoginTime	datetime	

Table Name : **WorkingTags**

TagId	bigint	
Status	bit	
Id	int	

Table Name : **WorkOrder**

WorkOrderNo	int	
CustId	int	
TagId	int	
StatusId	int	
Remarks	varchar	250
Updatedon	datetime	

OUTPUT DESIGN

Computer output is the most important and the direct source of information for the user. output design is very much dependent on the type of output and the nature of data and editing and totaling of data. The primary objective of output design is to arrange the data in a form most convenient to the user.

Outputs from computer system are primarily to communicate the results of the processing to the user. They are also used to provide permanent copy of results for later consultations. Once the output medium has been detailed specification of output document can be carried out.

Computer programmers are expected to have fairly concept of how to display computer output user to ready understand, efficient, intelligible, output design to improve relationships with the users and help decision making.

In this project the report is produced on the various formats, such as

- Reports produced based on Tag Wise
- Reports produced based on Customer Wise
- Reports produced based on Work Order wise
- Reports produced based on Date Wise

INTERFACE DESIGN

Once an operational user interface prototype has been created, it must be evaluated to determine whether it meets the needs of the user. Evaluation can span a formality spectrum that ranges from an informal “test drive,” in which a user provides impromptu feedback to a formally designed study that uses statistical methods for the evaluation of questionnaires completed by a population of end-users.

FROND END DESIGN

ASP.NET

Gain mature and powerful, built-in functionality, including a rich set of collection classes, networking support, multithreading support, string and regular expression classes, and broad support for XML, XML Schemas, XML namespaces, XSLT. Microsoft designed VB from the ground up to take advantage of its new .NET Framework. The .NET Framework is made up of four parts,

- The Common Language Runtime
- A Set Of Class Libraries
- A Set Of Programming Languages
- ASP.NET Environment

The .NET Framework is designed with three goals in mind. First, it was intended to make Windows applications much more reliable, while also providing an application with greater degree of security. Second, it was intended to simplify the development of Web applications and services that not only work in the traditional sense, but on mobile devices as well. Lastly, the framework was designed to provide a single set of libraries that would work with multiple languages. The .NET Framework is the infrastructure for the new Microsoft .Net Platform. The .NET Framework is a common environment for building, developing, and running Web applications and Web Services. The .NET Framework contains a common language runtime and common class libraries- like ADO.NET, ASP.NET and Windows Forms – to provide advanced standard services that can be integrated into a variety of computer systems. The .NET Framework provides a feature – rich application environment, simplified development and easy integration between a numbers of different development languages. The .NET Framework is

language neutral. Currently it supports ASP, C++, C# and Jscript (The Microsoft version of Java Script). Microsoft's Visual Studio.NET Framework.

Common Language Runtime:

One of the design goals of .NET Framework was to unify the runtime engines so that all developers could work with a set of runtime services. The .NET Framework's solution is called the Common Language Runtime (CLR). The CLR provides capabilities such as memory management, security, and robust error handling to any language that work with the .NET Framework. The CLR enables languages to inter operate with one another. Memory can be allocated by code written in one language and can be freed by code written in another language. Similarly, errors can be raised in one language and processed in another language.

NET Class Libraries:

The .NET Framework provides many classes that help developers re-use code. The .NET Class Libraries contain code for programming topics such as threading, file I/O, database support, XML parsing, and data structures such as stacks and queues. This entire class library is available o any programming languages that support the .NET Framework. Because all languages now support the same runtime, they can re-use any class that works with the .NET Framework. This means that any functionality available to one language will also be available to any other .NET language.

NET Programming Languages:

The .NET Framework provides a set of tools that help to build code that works with the .NET Framework. Microsoft provides a set of languages that are already .NET compatible. ASP.NET is one of those languages.

ASP.NET Environment:

Visual Basic were released by Microsoft to enable the creation of dynamic pages based on user input and interaction with a Web Site. ASP.NET improves upon the original VB by providing code-behind. ASP.NET Web services are XML – based services that are exposed on the Internet that can be SQL Server by other Web services and Web services clients

The application distributes the work between the local machine (the client) and the server (the backend), depending on the strengths of the client and server products. Client/Server systems are often very efficient because they minimize network traffic, and each portion of the application can be optimized for its particular function.

Language Support

- ASP .NET uses the new ADO .NET.
- ASP .NET supports full Visual Basic, not VBScript.
- ASP .NET supports C# (C sharp) and C++.
- ASP .NET supports JScript as before.

ASP .NET Controls

ASP .NET contains a large set of HTML controls. Almost all HTML elements on a page can be defined as ASP .NET control objects that can be controlled by scripts.

ASP .NET also contains a new set of object oriented input controls, like programmable list boxes and validation controls.

A new data grid control supports sorting, data paging, and everything you expect from a dataset control.

Event Aware Controls

All ASP .NET objects on a Web page can expose events that can be processed by ASP .NET code.

Load, Click and Change events handled by code makes coding much simpler and much better organized.

ASP .NET Components

ASP .NET components are heavily based on XML. Like the new AD Rotator, that uses XML to store advertisement information and configuration.

User Authentication

ASP .NET supports forms-based user authentication, including cookie management and automatic redirecting of unauthorized logins.

User Accounts and Roles

ASP .NET allows for user accounts and roles, to give each user (with a given role) access to different server code and executables.

High Scalability

Much has been done with ASP .NET to provide greater scalability.

Server to server communication has been greatly enhanced, making it possible to scale an application over several servers. One example of this is the ability to run XML parsers, XSL transformations and even resource hungry session objects on other servers.

Compiled Code

The first request for an ASP .NET page on the server will compile the ASP .NET code and keep a cached copy in memory. The result of this is greatly increased performance.

Easy Configuration

Configuration of ASP .NET is done with plain text files. Configuration files can be uploaded or changed while the application is running. No need to restart the server. No more metabase or registry puzzle.

Easy Deployment

No more server restart to deploy or replace compiled code. ASP .NET simply redirects all new requests to the new code.

Compatibility

ASP .NET is not fully compatible with earlier versions of ASP, so most of the old ASP code will need some changes to run under ASP .NET.

To overcome this problem, ASP .NET uses a new file extension ".aspx". This will make ASP .NET applications able to run side by side with standard ASP applications on the same server.

What is ADO.NET?

ADO.NET is a part of the .NET Framework

ADO.NET consists of a set of classes used to handle data access

ADO.NET is entirely based on XML

ADO.NET has, unlike ADO, no Recordset object

The Connection

After we import all the necessary namespaces, we're ready to connect to our database. Now, whether you implement the DataReader or Dataset, your initial database connection will still be as follows:

```
SqlConnection objConnect = new SqlConnection (Your Connection String);
objConnect.Open();
```

Above, we set up our SQLConnection Connection object with our database connection information, and then we opened it. Listed below are the common connection object methods we could work with:

Open - Opens the connection to our database

Close - Closes the database connection

Dispose - Releases the resources on the connection object. Used to force garbage collecting, ensuring no resources are being held after our connection is used. Incidentally, by using the Dispose method you automatically call the Close method as well.

State - Tells you what type of connection state your object is in, often used to check whether your connection is still using any resources.

Ex. if (ConnectionObject.State == ConnectionState.Open)

As far as opening a database connection goes, that's really the extent of it. Now we have to decide which object to use in order to achieve the end results you wish to present. We now have to choose whether to work with a Datareader or the Dataset. Let's begin by looking at the DataReader.

The DataReader

With classic ASP, when we needed a method for data retrieval, we'd use the appropriate data object and set its cursors to the task at hand. If we wanted a quick forward-only data read, we would set our Recordset's CursorType to adOpenForwardOnly and its LockType to adLockReadOnly (often referred to as a "fire- hose" cursor). Well, with .NET, all you need is the DataReader, which offers many features by which you can further tweak the efficiency of its output.

Command Object Methods

Now that we know what the DataReader does, there are numerous methods that can be used with it to achieve your specific goals. Here are some methods the DataReader works with through its Command object. The four Execute methods all pertain to an action performed by the Command object, wile the remaining methods are used to enhance the Command object's own functionality.

ExecuteReader	- Simply executes the SQL query against the database, using the Read() method to traverse through data, as illustrated below
ExecuteNonQuery	- Used whenever you work with SQL stored procedures with parameters, as illustrated in the Stored Procedures section below

ExecuteScalar - Returns a lightning fast single value as an object from your database

Ex. object val = Command.ExecuteScalar(); Then check if != null.

ExecuteXmlReader - Executes the SQL query against SQL Server only, while returning an XmlReader object.

Prepare - Equivalent to ADO's Command.Prepared = True property. Useful in caching the SQL command so it runs faster when called more than once.

Ex. Command.Prepare();

Dispose - Releases the resources on the Command object. Used to force garbage collecting, ensuring no resources are being held after our connection is used. Incidentally, by using the Dispose method you automatically call the Connection object's Close method as well.

Thus, after we establish our initial database connection all we need to do to retrieve data with a DataReader is to use the Command object to query the open database. The Command.ExecuteReader method used via the Command Object creates our DataReader.

```
SqlCommand objCommand = new SqlCommand(Sql String, objConnect);
SqlDataReader objDataReader = objCommand.ExecuteReader();
```

BACK END DESIGN

SQL Server:

SQL Server is such a rich and powerful that most people where to begin when they start using it.

SQL Server makes it easy for users even beginners to work with databases, you can Create Table, Edit data and use queries to find the data you want with very little effort, and SQL Server includes wizards that can do the work of designing data entry forms, reports and mailing labels for you.

SQL Server also makes it easy for develops to create applications. It includes as entire programming language, visual basic for applications, and its interface is so powerful that developers can create many custom applications without programming. SQL Server ‘ quires, reports and macros are powerful enough to do most of the work that use to require programming.

When users learn SQL Server that often find they have to wade through long discussions of power features to learn about the simple features they need to work with on their own data. The help system is so extensive and complex discussion of properties, expressions and other advanced features along with instructions on the basis of creating tables, queries, simple forms, reports and mailing labels, which average the user actually needs.

SQL Server has become the best selling database management programs because of its combination of power and easy of use. It is powerful enough that developers can use it to create entire applications. Yet it is easy enough to use that in a short time, beginners can learn to manage their own data with SQL Server.

In others database management programs, the term database is sometimes used to refer to tables that hold data. SQL Server uses the term more broadly. An SQL Server database consists of the tables that hold the data and all he related objects such as quires, forms and reports that are used to manage the data. Microsoft SQL Server will probably be an option on the programming menu, which you can select to start the program.

When you open a database, SQL Server displays the database window sometimes called as the database container, because it contains all the objects that make up the database.

There are two terms you should know, as they are almost always used when computerized databases are discussed. They are simply new names for familiar things.

A record is all the data on a single entity. For example, if have a list of names and address, each record includes the name and address of one person. In the sample table, where you have a list of products, each record includes the name and other details for one product.

A filed is one piece of data that appears up each record. For example, if you have a list of names and addresses the first name might be the first field, the last name might be the second field and the street address might be the third field. In the same table there are fields of product id, product name, supplier and there.

A table is made up of all the similar records you want to work on together. For example you might want to keep the names and address of all your friends in second table.

The simplest and most important type of query is a select query is a select query, which lets you select which data from a table is displayed. You can specify which fields

are displayed, enter criteria to specify which records are displayed and specify the sort order of these records.

Forms let you control how data is displayed on the screen. Macros let you automate and speed up your work; they are also used when you develop applications. A macro is a list of actions. Macros can save time for SQL Server users.

Multiple Databases:

Most Business applications database because there are many types of data that you cannot store effectively in a single table.

When businesses first begin to use computers they discovered they could eliminate repetitious data by breaking up some databases into multiple tables.

When a business kept records on paper, for example, the payroll department had a file with each employee's name, address, social security number unmanned data on the benefits department had a file with each employee's name, address, Social Security number and data on the benefits to which each was entitled.

The computer can looks up the name and address that go with each payroll record. It can retrieve data so quickly that it can display the name and address ROM one table and the payroll data from another table just as quickly as if they were in a single table.

You only have to enter the basic data once, though. When sometimes you only have to change the address in one table, and it will automatically be changed in all the forms used by all the departments.

Relational Database is not needed for very simple data, such as a mailing list, but they are needed for most business applications.

Internet Integration:

The SQL Server 2000 programming model is integrated with the Windows DNA architecture for developing Web applications, and SQL Server 2000 supports features such as English Query and the Microsoft Search Service to incorporate user-friendly queries and powerful search capabilities in Web application.

Scalability and Availability:

The same database engine can be used across platforms ranging from laptop computers running Microsoft Windows 98 through large, multiprocessor servers running Microsoft Windows 2000 Data center edition. SQL Server 2000 Enterprise Edition supports features such as federated servers, indexed views, and large memory support that allow it to scale to the performance levels required by the largest Websites.

Ease of installations, deployment and use:

SQL Server 2000 includes a set of administrative and development tools that improve upon the process of installing, deploying, managing, and using SQL Server across several sites. SQL Server 2000 also supports a standards-based programming model integrated with the Windows DNA, making the use of SQL Server databases and data warehouses a seamless part of building powerful and scalable systems. These features allow you to rapidly deliver SQL Server applications that customers can implement with a minimum of installation and administrative overhead.

Enterprise Manager is the main administrative console for SQL Server installations. It provides you with a graphical "birds-eye" view of all of the SQL Server installations on your network. You can perform high-level administrative functions that affect one or more servers, schedule common maintenance tasks or create and modify the structure of individual databases.

Query Analyzer offers a quick and dirty method for performing queries against any of your SQL Server databases. It's a great way to quickly pull information out of a database in response to a user request, test queries before implementing them in other applications, create/modify stored procedures and execute administrative tasks.

SQL Profiler provides a window into the inner workings of your database. You can monitor many different event types and observe database performance in real time. SQL Profiler allows you to capture and replay system "traces" that log various activities. It's a great tool for optimizing databases with performance issues or troubleshooting particular problems.

Service Manager is used to control the MS SQL Server (the main SQL Server process), MSDTC (Microsoft Distributed Transaction Coordinator) and SQL Server Agent processes. An icon for this service normally resides in the system tray of machines running SQL Server. You can use Service Manager to start, stop or pause any one of these services.

Data Transformation Services (DTS) provide an extremely flexible method for importing and exporting data between a Microsoft SQL Server installation and a large variety of other formats. The most commonly used DTS application is the "Import and Export Data" wizard found in the SQL Server program group.

Database

A database is similar to a data file in that it is a storage place for data. Like a data file, a database does not present information directly to a user; the user runs an application that accesses data from the database and presents it to the user in an understandable format.

Database systems are more powerful than data files. The data is more highly organized. In a well-designed database, there are no duplicate pieces of data that the user or application has to update at the same time. Related pieces of data are grouped together in a single structure or record, and relationships can be defined between these structures and records.

When working with data files, an application must be coded to work with the specific structure of each data file. In contrast, a database contains a catalog that applications use to determine how data is organized. Generic database applications can use the catalog to present users with data from different databases dynamically, without being tied to a specific data format.

Relational Database

There are different ways to organize data in a database but relational databases are one of the most effective. Relational database systems are an application of mathematical set theory to the problem of effectively organizing data. In a relational database, data is collected into tables (called relations in relational theory).

A table represents some class of objects that are important to an organization. For example, a company may have a database with a table for employees, another table for customers, and another for stores. Each table comprises columns and rows (attributes and tuples in relational theory). Each column represents some attribute of the object represented by the table. For example, an Employee table would typically have columns for first name, last name, employee ID, department, pay grade, and job title. Each row

represents an instance of the object represented by the table. For example, one row in the Employee table represents the employee who has employee ID 12345.

When organizing data into tables, you can usually find many different ways to define tables. Relational database theory defines a process, normalization, which ensures that the set of tables you define will organize your data effectively.

While SQL Server is designed to work as a server in a client/server network, it is also capable of working as a stand-alone database directly on the client. The scalability and ease-of-use features of SQL Server allow it to work efficiently on a client without consuming too many resources.

Structured Query Language (SQL)

To work with data in a database, you must use a set of commands and statements (language) defined by the DBMS software. There are several different languages that can be used with relational databases; the most common is SQL. Standards for SQL have been defined by both the American National Standards Institute (ANSI) and the International Standards Organization (ISO). Most modern DBMS products support the Entry Level of SQL-92, the latest SQL standard (published in 1992).

ALGORITHM USED

Login entry:

STEP-1: Select a new form.

STEP-2: Add controls two buttons, two textboxes, two labels and one Checkbox

STEP-3: Enter the name and password.

STEP-4: Click the sign in button to check the username and password.

STEP-5: Click the reset button to cancel the value and enter the new value.

User entry:

STEP-1: Select a new form.

STEP-2: The user module consists of general, login and communication information

STEP-4: The first name, last name entered in textbox.

STEP-5: The address is entered in multi line textbox.

STEP-6: The department and designation are entered in dropdown list.

STEP-7: The communication information consists of three textbox

STEP-8: The mail id, phone home, phone office are entered in a textbox

STEP-9: The login information consists of two textbox.

STEP-10: The username and password are entered in the textbox

STEP-11: The add button add the records to the database.

STEP-12: The view button view the records from the database

STEP-13: The delete button is used to delete the records from the Database

STEP-14: The edit button is used to edit the user records in the Database

Testing

TESTING

Definition of Testing:

Testing is a process of running software in an intention to find errors the software, which has been developed, has to be tested to prove its validity. Testing is considered to be the least creative phase of the whole cycle of system design. In the real sense it is the phase, which helps to bring out the creativity of the phases.

Testing program is the first step in the debugging process. Some people idea of the testing a program consists of running the program a few times to see what happens, each time using slightly different input. This process can succeed when we have a short program, but it is not effective for a long program. In any case, even for the simplest program the choice of test data is all-important.

A good testing suite is vital because the user must test all possible Execution paths inside the code in order to have any hope it will be bug free. Testing program is an art, not a science.

Testing of the system is done to ensure the integrity of the system. Testing is vita for the success of the project, which is the last stage of development. The aim of testing is to prove that the development system addresses the predefined processing requirements and will perform reliably and efficiently during time.

Testing is the process of creating a program with the explicit information of finding error that makes a program fail. Successful test finds the error yet undiscovered. As an additional benefit, testing demonstrates that a software function appear to be working according to the specifications.

Purposes of testing:

- ❖ To affirm the quality of the project.
- ❖ To find and eliminate any errors from previous stages.
- ❖ To validate the software and to eliminate the operation.
- ❖ Reliability of the system.

Types of testing:

The software, which has been developed, has to be tested to prove its validity. Testing is considered to be the least creative phase of the whole cycle of system design. In the real sense it is the phase, which helps to bring out the creativity of the phases.

The "Time Assessment for molding of casting" was tested along the following guidelines to prove its validity. It was tested using two techniques of software testing.

Unit Testing

In the lines of this strategy all the individual functions and modules were put to the test independently. By following this strategy all the error in coding were identified and corrected. This method was applied in combination with the white and Black box testing techniques to find the errors in each module.

Integration Testing

This testing strategy has two different approaches namely the Top-down approach in which the integration is carried out from the top level module to the bottom and the Bottom-up approach in which the integration is carried out from the low level module to the top.

The modules were tested using the Bottom-up approach by introducing stubs for the top-level functions. This test was used to identity the errors in the interfaces, the errors in passing the parameters between the functions and to correct them.

Validation Testing

The main aim of this testing is to verify that the software system does what it was designed for Alpha testing and Beta testing were conducted to ensure the validity of the system. A customer conducted the Alpha test at the developer's site. The end user of the software conducted the Beta test at one or more of the customer site.

System Testing

The idea of applying this strategy was to put the system to a series of different test whose primary purpose is to fully exercise the computer-based system. Although each test has a different purpose, all work to verify that all system elements have been properly integrated and perform allocated functions, security testing was conducted.

Acceptance testing

Acceptance testing involves planning and execution of functional tests, performance test and stress test to verify that the implemented system satisfies its requirements. Typically, acceptance test will incorporate test cases developed during unit testing and integration testing.

Additional test cases are added to achieve the desired level of functional, performance and stress testing of the entire system. Tools of special importance during acceptance testing include a test coverage analyzer, and a coding standard checker.

PROBLEM FACED

During this project faced lot of problems. Mainly the problem is to read the coding of the existing forms and understand how they are built that.

Most of the time machine, programmer, and module are not correctly retrieved from the database.

For the allocation, the module and machine are changed. At that time only the project name are not retrieved correctly.

- Very difficulty to maintain the Security
- It is tedious to update and correct the information in files.
- Preparation of reports is a time consuming task
- Possibility for any unauthorized person entering wrong Details.

FEATURE PLANS

The performance of any software product can be measured by its ability to satisfy the user needs, the speed at which it is done, its ability to integrate with the various functions and its ability to capture the errors and conveying the performance of the software is up to the satisfactory level.

Extensions can be done in the technology aspects by using data warehouse and data mining tools to adopt online databases for the end to end process which will help the network to function faster

Limitations of the software are again dependent on the economic feasibility of the user. If the user is provided with perfect and suitable configuration the software performance is still raised.

Always there is a need for improvement and enhancement for software. Even giants like Microsoft are also producing the enhanced versions even after developed an efficient product. This project work is feasible enough for amendments and modifications that may arrive in future.

References

1. *Kamal Acharya. School management system project report. Authorea. August 01, 2024. DOI: https://doi.org/10.22541/au.172254873.34023165/v1*

2. *Kamal Acharya. A CASE STUDY OF CINEMA MANAGEMENT SYSTEM PROJECT. Authorea. August 01, 2024. DOI: https://doi.org/10.22541/au.172254873.30191075/v1*

3. *Kamal Acharya. A CASE STUDY ON ONLINE TICKET BOOKING SYSTEM PROJECT. Authorea. August 01, 2024 DOI: https://doi.org/10.22541/au.172254872.26972790/v1*

4. *Kamal Acharya. Web chatting application project report management system. Authorea. August 01, 2024. DOI: https://doi.org/10.22541/au.172254871.18588592/v1*

5. *Kamal Acharya. RETAIL STORE MANAGEMENT SYSTEM PROJECT REPORT. Authorea. August 01, 2024. DOI: https://doi.org/10.22541/au.172254871.14590154/v1*

6. *Kamal Acharya. SUPERMARKET MANAGEMENT SYSTEM PROJECT REPORT. Authorea. August 01, 2024. DOI: https://doi.org/10.22541/au.172252491.19145062/v1*

7. *Kamal Acharya. SOCIAL MEDIA MANAGEMENT SYSTEM PROJECT REPORT. Authorea. August 01, 2024. DOI: https://doi.org/10.22541/au.172252491.11210579/v1*

8. *Kamal Acharya. Online music portal management system project report. Authorea. August 01, 2024. DOI: https://doi.org/10.22541/au.172252488.89734698/v1*

9. *Kamal Acharya. COLLEGE BUS MANAGEMENT SYSTEM PROJECT REPORT. Authorea. July 31, 2024. DOI: https://doi.org/10.22541/au.172245277.70798942/v1*

10. *Kamal Acharya. AUTOMOBILE MANAGEMENT SYSTEM PROJECT REPORT. Authorea. July 31, 2024. DOI: https://doi.org/10.22541/au.172245276.67982593/v1*

11. *Kamal Acharya. Ludo management system project report. Authorea. July 31, 2024. DOI: https://doi.org/10.22541/au.172243999.98091616/v1*

12. *Kamal Acharya. Literature online quiz system project report. Authorea. July 31, 2024 DOI: https://doi.org/10.22541/au.172243825.53562953/v1*

13. *Kamal Acharya. Avoid waste management system project. Authorea. July 29, 2024. DOI: https://doi.org/10.22541/au.172228528.85022205/v1*

14. *Kamal Acharya. CHAT APPLICATION THROUGH CLIENT SERVER MANAGEMENT SYSTEM PROJECT. Authorea. July 29, 2024. DOI: https://doi.org/10.22541/au.172228527.74316529/v1*

ABBREVIATION USED

- **CLR** - **C**ommon **L**anguage **R**untime
- **CTS** - **C**ommon **T**ype **S**ystem
- **SEH** - **S**tandard **E**xception **H**andling
- **AJAX** - **A**synchronous **J**avaScript and **X**ml

CONCLUSION

The Asset Tracking process for the testers is successfully done in this project. any Software concern can use this project. This project is more effective and efficient. The special feature of this project is easy way to communicate between the Programmer (developers) and the testing team members. Almost all needs for maintaining a scheduling are fully implemented in this project. This Project is used by three types of users such as administrator, developers, testing team members of a software concern can use it. In future, it should be enhanced. This project will satisfy the needs of users who are using it.

The various processes of the construction of the new system have been done. The customized programs are effective and easy to handle, with good help and browse Facility. The necessary information is allowed to feed into the output screen. The Customized modules satisfy the user needs.

An inconvenience of this chat is that it is possible for several users to share a single name at different times. In the next part, it will be fixed by implementing registration mechanism.

The system is done with insight into the necessary modification that may require in the future. Hence the system can be maintained successfully without much rework.

The login operation will complete successfully only if no active users have the same name. But there are also possible situations where a user is disconnected from the network or closed their chat by mistake. In these cases, a mechanism is required to repeat user authorization.